Making ML models fairer through explanations: the case of LimeOut *

Guilherme Alves, Vaishnavi Bhargava, Miguel Couceiro, and Amedeo Napoli

Université de Lorraine, CNRS, Inria N.G.E., LORIA, F-54000 Nancy, France
{guilherme.alves-da-silva,miguel.couceiro,amedeo.napoli}@loria.fr
vaishnavi.bhargava2605@gmail.com

Abstract. Algorithmic decisions are now being used on a daily basis, and based on Machine Learning (ML) processes that may be complex and biased. This raises several concerns given the critical impact that biased decisions may have on individuals or on society as a whole. Not only unfair outcomes affect human rights, they also undermine public trust in ML and AI. In this paper we address fairness issues of ML models based on decision outcomes, and we show how the simple idea of "feature dropout" followed by an "ensemble approach" can improve model fairness. To illustrate, we will revisit the case of "LimeOut" that was proposed to tackle "process fairness", which measures a model's reliance on sensitive or discriminatory features. Given a classifier, a dataset and a set of sensitive features, LimeOut first assesses whether the classifier is fair by checking its reliance on sensitive features using "Lime explanations". If deemed unfair, LimeOut then applies feature dropout to obtain a pool of classifiers. These are then combined into an ensemble classifier that was empirically shown to be less dependent on sensitive features without compromising the classifier's accuracy. We present different experiments on multiple datasets and several state of the art classifiers, which show that LimeOut's classifiers improve (or at least maintain) not only process fairness but also other fairness metrics such as individual and group fairness, equal opportunity, and demographic parity.

Keywords: Fairness metrics · Feature importance · Feature-dropout · Ensemble classifier · LIME explanations

1 Introduction

Algorithmic decisions are now being used on a daily basis and obtained by Machine Learning (ML) processes that may be rather complex and opaque. This raises several concerns given the critical impact that such decisions may have on

* This research was partially supported by TAILOR, a project funded by EU Horizon 2020 research and innovation programme under GA No 952215, and the Inria Project Lab "Hybrid Approaches for Interpretable AI" (HyAIAI)

individuals or on society as a whole. Well known examples include the classifiers which are used to predict the credit card defaulters, including multiple other datasets which may impact the government decisions. These prevalent classifiers are generally known to be biased to certain minority or vulnerable groups of society, which should rather be protected. Most of the notions of fairness thus focus on the outcomes of the decision process [15, 16]. They are inspired by several anti-discrimination efforts that aim to ensure that unprivileged groups (e.g. racial minorities) should be treated fairly. Such issues can be addressed by looking into fairness individually [15] or as a group [15, 16]. Actually, earlier studies [18, 17] consider individual and group fairness as conflicting measures, and some studies tried to find an optimal trade-off between them. In [3] the author argues that, although apparently conflicting, they correspond to the same underlying moral concept, thus providing a broader perspective and advocating an individual treatment and assessment based on a case-by-case analysis.

The authors of [8, 7] provide yet another noteworthy perspective of fairness, namely, *process fairness*. Rather than focusing on the outcome, it deals with the process leading to the outcome. In [2] we delivered a potential solution to deal with process fairness in ML classifiers. The key idea was to use an explanatory model, namely, LIME [14] to assess whether a given classifier was fair by measuring its reliance on salient or sensitive features. This component was then integrated in a human-centered workflow called *LimeOut*, that receives as input a triple (M, D, F) of a classifier M, a dataset D and a set F of sensitive features, and outputs a classifier M_{final} less dependent on sensitive features without compromising accuracy. To achieve both goals, LimeOut relies on feature dropout to produce a pool of classifiers that are then combined through an ensemble approach. Feature dropout receives a classifier and a feature a as input, and produces a classifier that does not take a into account. This preliminary study [2] showed the feasibility and the flexibility of the simple idea of feature dropout followed by an ensemble approach to improve process fairness. However, the empirical study of [2] was performed only on two families of classifiers (logistic regression and random forests) and carried out on two real-life datasets (Adult and German Credit Score). Also, it did not take into account other commonly used fairness measures. Moreover, in a recent study [6], Dimanov *et al.* question the trustfulness of certain explanation methods when assessing model fairness. In fact, they present a procedure for modifying a pre-trained model in order to manipulate the outputs of explanation methods that are based on feature importance (FI). They also observed minor changes in accuracy and that, even though the pre-trained model was deemed fair by some FI based explanation methods, it may conceal unfairness with respect to other fairness metrics.

This motivated us to revisit *LimeOut*'s framework to perform a thorough analysis that follows the tracks of [6] and extends the empirical study of [2] in several ways: (i) we experiment on many other datasets (e.g., HDMA dataset, Taiwanese Credit Card dataset, LSAC) , (ii) we make use of a larger family of ML classifiers (that include AdaBoost, Bagging, Random Forest (RF), and Logistic Regression (LR)), and (iii) we evaluate *LimeOut*'s output classifiers with

respect to a wide variety fairness metrics, namely, disparate impact (DI), disparate mistreatment or equal opportunity (EO), demographic parity (DP), equal accuracy (EA), and predictive equality (PQ). As it will become clear from the empirical results, the robustness of *LimeOut*'s to different fairness view points is once again confirmed without compromising accuracy.

The paper is organised as follows. After recalling Lime explanations and various fairness measures in Subsections 2.1 and 2.2, respectively, we briefly describe *LimeOut*'s workflow in Subsection 2.3. We then present in Section 3 an extended empirical study following the tracks of [2] and the recent study [6]. First we quickly describe the datasets used (Subsection 3.1) and the classifiers employed (Subsection 3.2). We then present the empirical results and the various assessments with respect to the different fairness metrics considered in Subsection 2.2. We conclude the paper in Section 4 with some final remarks on ongoing work and perspectives of future research.

2 Related Work

In this section, we briefly recall LIME (Subsection 2.1), recall the different metrics used to measure model fairness (Subsection 2.2) and revisit *LimeOut*'s framework (Subsection 2.3).

2.1 LIME - Explanatory Method

Recall that LIME explanations [14] (Local Interpretable Model Agostic Explanations) take the form of surrogate linear models, that locally mimic the behavior of a ML model. Essentially, it tries to find the best possible linear model (i.e. explanation model) which fits the prediction of ML model of a given instance and it's neighbouring points (see below).

Let $f : \mathbb{R}^d \to \mathbb{R}$ be the function learned by a classification or regression model over training samples. LIME's workflow can be described as follows. Given an instance x and its ML prediction $f(x)$, LIME generates neighbourhood points by perturbing x and gets their corresponding predictions. These neighbouring points z are assigned weights based on their proximity to x, using the following equation:

$$\pi_x(z) = e^{(\frac{D(x,z)^2}{\sigma^2})},$$

where $D(x, z)$ is the Euclidean distance between x and z, and σ is the hyper parameter (kernel-width). LIME then learns the weighted linear model g over the original and neighbourhood points, and their respective predictions, by solving the following optimization problem:

$$g = argmin_{g \in \mathcal{G}}\ \mathcal{L}(f, g, \pi_x(z)) + \Omega(g),$$

where $L(f, g, \pi_x(z))$ is a measure of how unfaithful g is in approximating f in the locality defined by $\pi_x(z)$. $\Omega(g)$ measures the complexity of g (regularization

term). In order to ensure both interpretability and local faithfulness, LIME minimizes $L(f, g, \pi_x(z))$ while enforcing $\Omega(g)$ to be small in order to be interpretable by humans. The obtained explanation model g is of the form

$$g(x) = \hat{\alpha}_0 + \sum_{1 \leq i \leq d'} \hat{\alpha}_i x[i],$$

where $\hat{\alpha}_i$ represents the contribution or importance of feature $x[i]$. Figure 1 presents the explanation of LIME for the classification of an instance from the Adult dataset. For instance, the value "Capital Gain" ≤ 0.0 contributes 0.29 to the class $\leq 50K$, whereas the value "Relationship" $= Husband$ contributes 0.15 to the class $> 50K$.

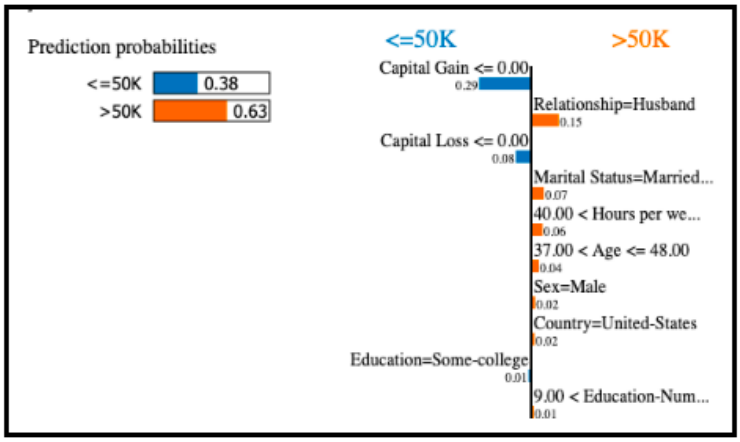

Fig. 1: LIME explanation in case of Adult dataset

2.2 Model Fairness

Several metrics have been proposed in the literature in order to assess ML model's fairness. Here we recall some of the most used ones.

- **Individual Fairness**[1] [4] imposes that the instances/individuals belonging to different sensitive groups, but with similar non-sensitive attributes must receive equal decision outcomes.
- **Disparate Impact**[2] (DI) [5] is rooted in the desire for different sensitive demographic groups to experience similar rates of positive decision outcomes ($\hat{y} = pos$). Given the ML model, $\hat{y}$ represents the predicted class. It compares

[1] It is also referred to as *disparate treatment* or *predictive parity*
[2] It is also referred to as *group fairness*

two groups of the population based on a sensitive feature: the privileged ($priv$) and the unprivileged (unp) groups. For instance, if we consider race as sensitive feature, white people can be assigned as privileged and non-white people as unprivileged group.

$$DI = \frac{P(\hat{y} = pos|D = unp)}{P(\hat{y} = pos|D = priv)}$$

- **Equal Opportunity**[3] **(EO)** [16] proposes different sensitive groups to achieve similar rates of error in decision outcomes. It is computed as the difference in recall scores ($\frac{TP_i}{TP_i+FN_i}$, where TP_i is true positive and FN_i is false negative for a particular group i) between the unprivileged and privileged groups.

$$EO = \frac{TP_{unp}}{TP_{unp} + FN_{unp}} - \frac{TP_{priv}}{TP_{priv} + FN_{priv}}$$

- **Process Fairness**[4] [8, 7] deals with the process leading to the prediction and keeps track of input features used by the decision model. In other words, the process fairness deals at the algorithmic level and ensures that the algorithm does not use any sensitive features while making a prediction.
- **Demographic Parity (DP)** [9] the difference in the predicted positive rates between the unprivileged and privileged groups.

$$DP = P(\hat{y} = pos|D = unp) - P(\hat{y} = pos|D = priv)$$

- **Equal Accuracy (EA)** [9] the difference in accuracy score ($\frac{TP_i+TN_i}{P_i+N_i}$, where TN_i is true negative of a particular group i) between unprivileged and privileged groups.

$$EA = \frac{TP_{unp} + TN_{unp}}{P_{unp} + N_{unp}} - \frac{TP_{priv} + TN_{priv}}{P_{priv} + N_{priv}}$$

- **Predictive Equality (PE)** which is defined as the difference in false positive rates ($\frac{FP_i}{FP_i+TP_i}$, where FP_i is false positive for a particular group i) between unprivileged and privileged groups. Formally,

$$PE = \frac{FP_{unp}}{FP_{unp} + TP_{unp}} - \frac{FP_{priv}}{FP_{priv} + TP_{priv}}.$$

In this paper we follow the same empirical setting of [6] and [2] and, hence, will focus mainly on disparate impact, equal opportunity, process fairness, demographic parity and equal accuracy.

[3] It is also referred to as *disparate mistreatment*
[4] It is also referred to as *procedural fairness*

2.3 *LimeOut*'s framework

In this subsection, we briefly describe *LimeOut*'s framework, which essentially consists of two main components: $\text{LIME}_{\text{Global}}$ and $\text{ENSEMBLE}_{\text{Out}}$. Given an input (M, D, F), where M is a classifier, D is a dataset, and F is a list of sensitive features, *LimeOut* first employs a "global variant" of LIME ($\text{LIME}_{\text{Global}}$) to assess the contribution (importance) of each feature to the classifier's outcomes. For that, $\text{LIME}_{\text{Global}}$ uses submodular pick to select instances with diverse and non-redundant explanations [14], and which are then aggregated to provide a global explanations (see [2]). The final output of $\text{LIME}_{\text{Global}}$ is thus a list of the k most important features[5].

If the k most important feature contain at least two sensitive features in F, then the model is deemed unfair (or biased), and the second component $\text{ENSEMBLE}_{\text{Out}}$ is deployed. Essentially, $\text{ENSEMBLE}_{\text{Out}}$ applies feature dropout on the sensitive features that are among the k most important features, each of which giving rise to a classifier obtained from M by removing that feature. thus resulting in a pool of classifiers. $\text{ENSEMBLE}_{\text{Out}}$ then constructs an ensemble classifier M_{final} through a linear combination of the pool's classifiers.

More precisely, if $\text{LIME}_{\text{Global}}$ outputs $a_1, a_2, \ldots, a_k$ as the k most important features, in which $a_{j_1}, a_{j_2}, \ldots, a_{j_i}$ are sensitive, then *LimeOut* trains $i+1$ classifiers: M_t after removing a_{j_t} from the dataset, for $t = 1, \ldots, i$, and M_{i+1} after removing all sensitive features $a_{j_1}, a_{j_2}, \ldots, a_{j_i}$. The ensemble classifier M_{final} is then defined as the "average" of these $i+1$ classifiers, i.e., by the rule: for an instance x and a class C,

$$P_{M_{final}}(x \in C) = \frac{\sum_{t=1}^{i+1} P_{M_t}(x \in C)}{i+1}.$$

The empirical studies carried out in [2] showed that this ensemble classifier obtained by *LimeOut* is fairer with respect to process fairness than the input model M, without compromising (or even improving) M's accuracy.

3 Empirical study

In this section, we first describe in Subsection 3.1 the datasets that we used in our experiments, and we briefly present in Subsection 3.2 the empirical setup. We then discuss our results from different points of view. In Subsection 3.3 we report on the improved accuracy of *LimeOut*'s classifiers using different models and on the various datasets considered. We will then assess the fairness of *LimeOut*'s classifiers in Subsection 3.3: first on process fairness and then on the remaining metrics of Subsection 2.2.

3.1 Datasets

Experiments were conducted using five datasets. All datasets share common characteristics that allow us to run our experiments: a binary target feature and

[5] In [2] k was set to 10.

the presence of sensitive features. Table 1 summarizes basic information about these datasets. The details concerning each dataset are presented as follows.

Table 1: Datasets employed in the experiments.

Dataset	# features	# sensitive	# instances
Adult	14	3	32561
German	20	3	1000
HMDA	28	3	92793
Default	23	3	30000
LSAC	11	2	26551

Adult. This dataset is available on UCI repository[6]. The target variable indicates whether a person earns more then 50k dollars per year. The goal is to predict the target feature based on census data. In this dataset, we considered as sensitive features: "Marital Status", "Race", and "Sex".

German. This is also a dataset available on UCI repository[7]. The task is to predict if an applicant has a high credit risk. In other words, if an applicant is likely to pay back his loan. We considered as sensitive features: "statussex", "telephone", and "foreign worker".

HMDA. The *Home Mortgage Disclosure Act* (HMDA)[8] aims to help identifying possible discriminatory lending practices. This public data about home mortgage contains information about the applicant (demographic information), the lender (name, regulator), the property (type of property, owner occupancy, census tract), and the loan (loan amount, type of loan, loan purpose). Here, the goal is to predict whether a loan is "high-priced", and the features that are considered sensitive are "sex", "race", and "ethnicity".

Default. This dataset is also a dataset available on the UCI repository[9]. The goal is to predict the probability of default payments using data from Taiwanese credit card users, e.g., credit limit, gender, education, marital status, history of payment, bill and payment amounts. We consider as sensitive features in this dataset: "sex" and "marriage".

LSAC. The *Law School Admissions Council* (LSAC)[10] dataset contains information about approx. 27K students through law school, graduation, and sittings for bar exams. This information was collected from 1991 through 1997, and it describes students' gender, race, year of birth (`DOB_yr`), full-time status, family income, Law School Admission Test score (`lsat`), and academic performace (undegraduate GPA (`ugpa`), standardized overall GPA (`zgpa`), standard-

[6] http://archive.ics.uci.edu/ml/datasets/Adult
[7] https://archive.ics.uci.edu/ml/datasets/statlog+(german+credit+data)
[8] https://www.consumerfinance.gov/data-research/hmda/
[9] https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients
[10] http://www.seaphe.org/databases.php

ized 1st year GPA (`zfygpa`), weighted index using 60% of LSAT and 40% of `ugpa` (`weighted_lsat_ugpa`)). Here, the goal is to predict whether a law student passes in the bar exam. In this dataset, features that could be considered sensitive are "race" and "sex".

3.2 Empirical Setup

To perform our experiments[11], we split each dataset into 70% training set and 30% testing. As the datasets are imbalanced, we used Synthetic Minority Oversampling Technique (SMOTE[12]) over training data to generate the samples synthetically. We trained original and ensemble models on the balanced (augmented) datasets using Scikit-learn implementations [12] of the following five algorithms: *AdaBoost* (ADA), *Bagging*, *Random Forest* (RF), and *Logistic Regression* (LR). For ADA, Bagging, RF, and LR we kept the default parameters of Scikit-learn documentation[13].

3.3 Accuracy Assessment

Table 2 shows the average accuracy obtained in all experiments. We repeated the same experiment 10 times. For each dataset, we indicate the average accuracy of the original model ("Original") and the average accuracy of the *LimeOut* ensemble model (line "*LimeOut*"). Our analysis is based on the comparison between the accuracy of the original and the ensemble models. Since we drop sensitive features, it is expected that the accuracy of model decreases. However, it is evident that *LimeOut* ensemble models maintain the level of accuracy, even though sensitive features were dropped out.

We notice a slight improvement in the accuracy of the ensemble models when we use Bagging over German, Adult and Default datasets. Although in some cases we notice a difference between original and ensemble models, in all scenarios the difference is statistically negligible.

3.4 Fairness Assessment

We now assess model fairness with respect to two points of view, namely, in terms of *process fairness* and in terms of various *fairness metrics*.

Process Fairness. In this section we analyze the impact of feature dropout and the dependence on sensitive features. We employ $\text{LIME}_{\text{Global}}$ to compute feature contributions and build the list of the most important features. Instead of providing the lists of feature contributions for all combinations of datasets

[11] The gitlab repository of LimeOut can be found here: https://gitlab.inria.fr/orpailleur/limeout

[12] https://machinelearningmastery.com/threshold-moving-for-imbalanced-classification/

[13] We used version 0.23.1 of Scikit-learn.

Table 2: Average accuracy assessment, where *LimeOut* stands for the ensemble model built by our proposed framework. Numbers in parentheses indicate standard deviation. No accuracy values are reported on the HMDA dataset for logistic regression, and on the Default dataset for random forest and logistic regression, since in each of these cases the original model was deemed fair.

		ADA	Bagging	RF	LR
German	Original	0.757 (0.015)	0.743 (0.019)	**0.772** (0.016)	0.769 (0.021)
	LimeOut	0.765 (0.014)	0.755 (0.021)	0.769 (0.016)	0.770 (0.021)
Adult	Original	**0.855** (0.003)	0.841 (0.002)	0.808 (0.007)	0.845 (0.004)
	LimeOut	0.856 (0.003)	0.849 (0.002)	0.808 (0.004)	0.849 (0.004)
HMDA	Original	0.879 (0.001)	**0.883** (0.001)	0.882 (0.001)	0.878 (0.001)
	LimeOut	0.880 (0.001)	0.884 (0.000)	0.884 (0.000)	-
LSAC	Original	0.857 (0.003)	**0.861** (0.002)	0.852 (0.002)	0.820 (0.006)
	LimeOut	0.859 (0.002)	0.866 (0.002)	0.859 (0.002)	0.822 (0.005)
Default	Original	**0.817** (0.003)	0.804 (0.003)	0.807 (0.003)	0.779 (0.004)
	LimeOut	0.817 (0.003)	0.812 (0.002)	-	-

and classifiers, for each dataset, we select the classifier that provides the highest accuracy, as we did in Subsection 3.3.

We thus look at the explanations obtained from $\text{LIME}_{\text{Global}}$ for these selected combinations. Tables 3, 4, 5, 6 and 7 present the list of most important features for these datasets. In all cases, we can notice that *LimeOut* decreases the dependence on sensitive features. In other words, the ensemble models provided by our framework have less sensitive features in the list of most important features. Also, LIME explanations show that the remaining sensitive features (the ones that appeared in the list of the ensemble model) contributed less to the global prediction compared to the original model.

For all datasets we used $k = 10$, except for the HMDA dataset. Indeed, in the latter case we took $k = 15$ (Table 5). This is due to the fact that all models were considered fair by *LimeOut* if only the first 10 important features were taken into account. We thus decided to investigate whether considering more features would show a different result, as it turned out to be the case when applying Bagging on HMDA.

Fairness Metrics In this section, we assess fairness using the fairness metrics introduced in Section 2. We compute fairness metrics using IBM AI Fairness 360 Toolkit[14] [1]. Our goal is to have a different perspective on the fairness of *LimeOut* ensemble models since we only assessed fairness by using LIME explanations. In this analysis, we compare the original and ensemble models for each combination of classifier and sensitive feature.

[14] https://github.com/Trusted-AI/AIF360

Table 3: LIME explanations in the form of pairs feature/contribution for the original AdaBoost model and the ensemble variant (*LimeOut*'s output) on the on Adult dataset.

Original

Feature	Contrib.
CapitalGain	-18.449067
CapitalLoss	-4.922207
Hoursperweek	3.297749
Workclass	-0.997601
fnlwgt	-0.890244
MaritalStatus	0.873829
Sex	0.694676
Education-Num	-0.603877
Relationship	0.277705
Occupation	0.173059

Ensemble

Feature	Contrib.
CapitalGain	-19.147673
CapitalLoss	-9.682837
Hoursperweek	1.173417
fnlwgt	0.974685
Workclass	-0.423646
Education-Num	-0.259837
Sex	-0.244728
Country	-0.162728
Education	0.127105
Age	-0.124858

Table 4: LIME explanations of RF on German dataset.

Original

Feature	Contrib.
foreignworker	2.664899
otherinstallmentplans	-1.354191
housing	-1.144371
savings	0.984104
property	-0.648104
purpose	-0.415498
existingchecking	0.371415
telephone	0.311451
credithistory	0.263366
duration	-0.223288

Ensemble

Feature	Contrib.
otherinstallmentplans	-1.487604
housing	-1.089726
savings	0.679195
duration	-0.483643
foreignworker	0.448643
property	-0.386355
credithistory	0.258375
job	-0.252046
existingchecking	-0.21358
residencesince	-0.138818

Table 5: LIME explanation of Bagging on HMDA dataset.

Original

Feature	Contrib.
derived_loan_product_type	4.798847
balloon_payment_desc	4.624029
intro_rate_period	4.183828
loan_to_value_ratio	2.824717
balloon_payment	2.005847
prepayment_penalty_term	0.683618
reverse_mortgage	-0.659169
applicant_age_above_62	0.532331
derived_ethnicity	-0.409255
co_applicant_age_above_62	-0.333838
property_value	-0.326801
derived_race	-0.318802
applicant_age	-0.304565
loan_term	0.270951
negative_amortization	-0.229379

Ensemble

Feature	Contrib.
derived_loan_product_type	6.457707
balloon_payment_desc	5.054243
intro_rate_period	4.638744
balloon_payment	1.512304
prepayment_penalty_term	-1.267424
interest_only_payment	0.777766
loan_to_value_ratio	0.704758
negative_amortization_desc	0.61936
reverse_mortgage_desc	0.508204
interest_only_payment_desc	-0.393068
applicant_credit_score_type_desc	-0.379852
negative_amortization	-0.353717
applicant_age_above_62	0.349847
property_value	-0.316311
applicant_credit_score_type	-0.192114

Figures 2, 3 and 4 show values for all fairness metrics in each graphic. Red points indicate the values for *LimeOut* ensemble models while blue points indicate values for original models. The dashed line is the reference for a fair model (optimal value), i.e., 0 for all metrics except DI where the optimal is 1.

Results for the German dataset are depicted in Figure 2. It is evident that *LimeOut* produces ensemble models that are fairer according to metrics DP and

Table 6: LIME explanations of AdaBoost on Default dataset.

Original		Ensemble	
Feature	**Contrib.**	**Feature**	**Contrib.**
PAY_0	0.014194	PAY_2	-0.024354
MARRIAGE	-0.013986	PAY_0	0.008862
PAY_2	-0.013513	PAY_5	0.008729
PAY_6	-0.011724	PAY_AMT6	-0.00566
PAY_AMT1	0.011664	LIMIT_BAL	-0.003584
PAY_AMT6	0.008088	BILL_AMT2	0.00329
PAY_AMT2	0.007735	PAY_6	-0.00307
PAY_3	0.00735	AGE	-0.002058
EDUCATION	0.0032	PAY_AMT1	0.001592
SEX	0.000732	PAY_3	-0.001492

Table 7: LIME explanations of Bagging on LSAC dataset.

Original		Ensemble	
Feature	**Contrib.**	**Feature**	**Contrib.**
isPartTime	-12.588169	isPartTime	-9.294158
race	-3.943962	cluster_tier	-3.464014
cluster_tier	-1.873394	zgpa	2.835836
DOB_yr	-1.235803	family_income	-1.292526
zgpa	-0.71457	DOB_yr	-0.923861
zfygpa	0.314865	**race**	-0.895484
ugpa	0.123805	zfygpa	0.238397
family_income	-0.08999	weighted_lsat_ugpa	0.060846
lsat	-0.07596	ugpa	-0.055593
sex	-0.068117	**sex**	-0.041478

EQ. Red points are closer to zero compared to blue points, which means that *LimeOut* ensemble models are fairer than pre-trained models. We can also notice general improvement on DI. However, we observe that the only problematic sensitive feature is "foreignworker", where no improvement is observed. For all other sensitive features, we observe an improved fairness behaviour. In a few cases, the differences are negligible, which indicates that *LimeOut* either improves or at least maintains the fairness metrics.

Figure 3 shows the results on fairness metrics for the Adult dataset. In this dataset, *LimeOut* ensemble models keep values of all metrics in almost scenarios. We only see a deterioration of fairness when we compute EQ for Logistic Regression focuses on marital status. This behaviour means that *LimeOut* at least maintain the value of fairness metrics when it reduces the dependence on sensitive features, but it cannot ensure fairness metrics closer to 0.

The fairness metrics for LSAC dataset are depicted in Figure 4. For this dataset, most of results indicate that *LimeOut* maintains the fairness measurements. We can observe some exceptions, for instance, "race" with Bagging on PE and EQ, where an improvement is observed. This behaviour can indicate that, even if *LimeOut*'s ensemble outputs are in general less dependent on sensitive features, for some datasets a weighted aggregation of pool classifiers should be employed (Section 2.3). For HMDA and Default datasets we observed a similar behaviour even though lesser classifiers were deemed unfair. The results for these two latter datasets are presented in the Appendix A and the fairness met-

rics show a rather fair behaviour of the few models that were deemed unfair by *LimeOut*.

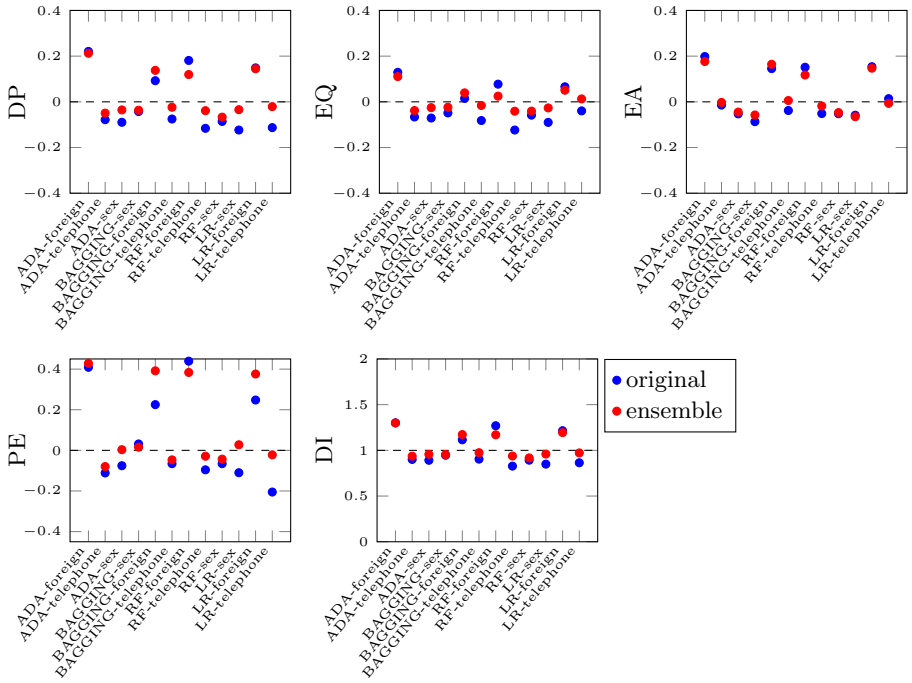

Fig. 2: Fairness metrics for German Credit Score Dataset

4 Conclusion and Future Work

In this paper we revisited *LimeOut*'s framework that uses explanation methods in order to assess model fairness. *LimeOut* uses LIME explanations, and it receives as input a triple (M, D, F) of a classifier M, a dataset D and a set of "sensitive" features F, and outputs a fairer classifier M_{final} in the sense that it is less dependent on sensitive features without compromising the model's accuracy. We extended the empirical study of [2] by including experiments of a wide family of classifiers on various and diverse datasets on which fairness issues naturally appear. These new experiments reattested what was empirically shown in [2], namely, that *LimeOut* improves process fairness without compromising accuracy.

However, the authors of [6] raised several concerns in such an approach based on explanation methods that use feature importance indices to determine model fairness since they conceal other forms of unfairness. This motivated us to deepen

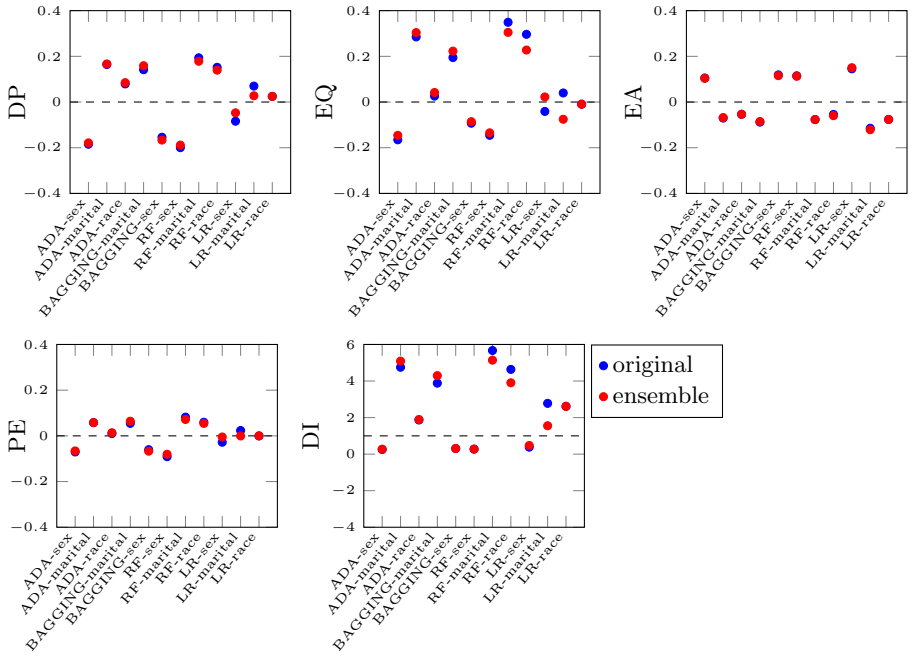

Fig. 3: Fairness metrics for Adult Dataset.

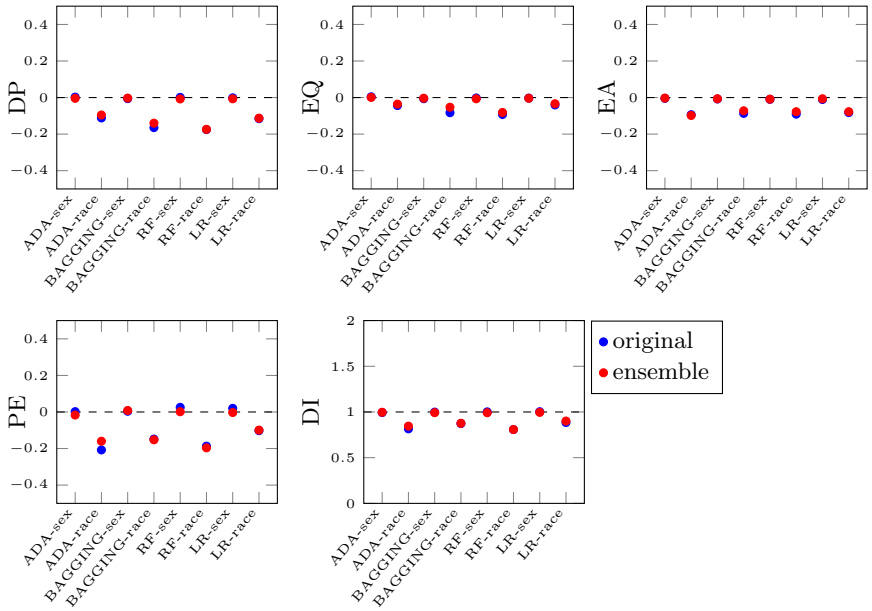

Fig. 4: Fairness metrics for LSAC Dataset.

the thorough analysis of *LimeOut* to evaluate the model outcomes of *LimeOut* with respect to several well known fairness metrics. Our results show consistent improvements in most metrics with a very few exceptions that will be investigated in more detail. Also, we have already adapted *LimeOut* to other data types and different explanatory models such as SHAP [11] and Anchors [13]. However, the construction of global explanations like [10] should be thoroughly explored. Also, the aggregation rule to produce classifier ensembles should be improved in order take into account classifier weighting, as well as other classifiers resulting from the removal of different subsets of sensitive features (here we only considered the removal of one or all features). Finally, we took a human and context-centered approach for identifying sensitive features in a given use-case. There is hope to automating this task while taking into account domain knowledge and using statistical dataset characteristics and utility-based approaches to quantify sensitivity. This will be the topic of a follow up contribution.

References

1. Bellamy, R.K.E., *et al.*: AI Fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. ArXiv **abs/1810.01943** (2018)
2. Bhargava, V., Couceiro, M., Napoli, A.: LimeOut: An Ensemble Approach To Improve Process Fairness. In: ECML PKDD Int. Workshop XKDD (2020)
3. Binns, R.: On the apparent conflict between individual and group fairness. In: FAT'20. pp. 514–524
4. Chouldechova, A.: Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. Big data **5**(2), 153–163 (2017)
5. Cynthia, D., *et al.*: Fairness through awareness. In: Innovations in Theoretical Computer Science. pp. 214–226. ACM (2012)
6. Dimanov, B., *et al.*: You shouldn't trust me: Learning models which conceal unfairness from multiple explanation methods. In: ECAI'20. pp. 2473–2480
7. Grgić-Hlača, N., *et al.*: Beyond distributive fairness in algorithmic decision making: Feature selection for procedurally fair learning. In: AAAI'18. pp. 51–60
8. Grgic-Hlaca, N., *et al.*: The case for process fairness in learning: Feature selection for fair decision making. In: NIPS Symposium on Machine Learning and the Law
9. Hardt, M., *et al.*: Equality of opportunity in supervised learning. In: NIPS'16
10. van der Linden, I., Haned, H., Kanoulas, E.: Global aggregations of local explanations for black box models. ArXiv **abs/1907.03039** (2019)
11. Lundberg, S.M., Lee, S.: A unified approach to interpreting model predictions. In: NIPS'17. pp. 4765–4774
12. Pedregosa, F., *et al.*: Scikit-learn: Machine learning in Python. JMLR **12**, 2825–2830 (2011)
13. Ribeiro, M.T., Singh, S., Guestrin, C.: Anchors: High-precision model-agnostic explanations. In: AAAI'18. pp. 1527–1535
14. Ribeiro, M.T., Singh, S., Guestrin, C.: "why should i trust you?": Explaining the predictions of any classifier. In: SIGKDD'16. pp. 1135–1144
15. Speicher, T., *et al.*: A unified approach to quantifying algorithmic unfairness: Measuring individual & group unfairness via inequality indices. In: SIGKDD'18. pp. 2239–2248

16. Zafar, M.B., *et al.*: Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In: WWW'17. pp. 1171–1180 (2017)
17. Zafar, M.B.e.: Fairness constraints: Mechanisms for fair classification. In: AISTATS'17. pp. 962–970
18. Zemel, R.e.: Learning fair representations. In: ICML'13. pp. 325–333

A Appendix

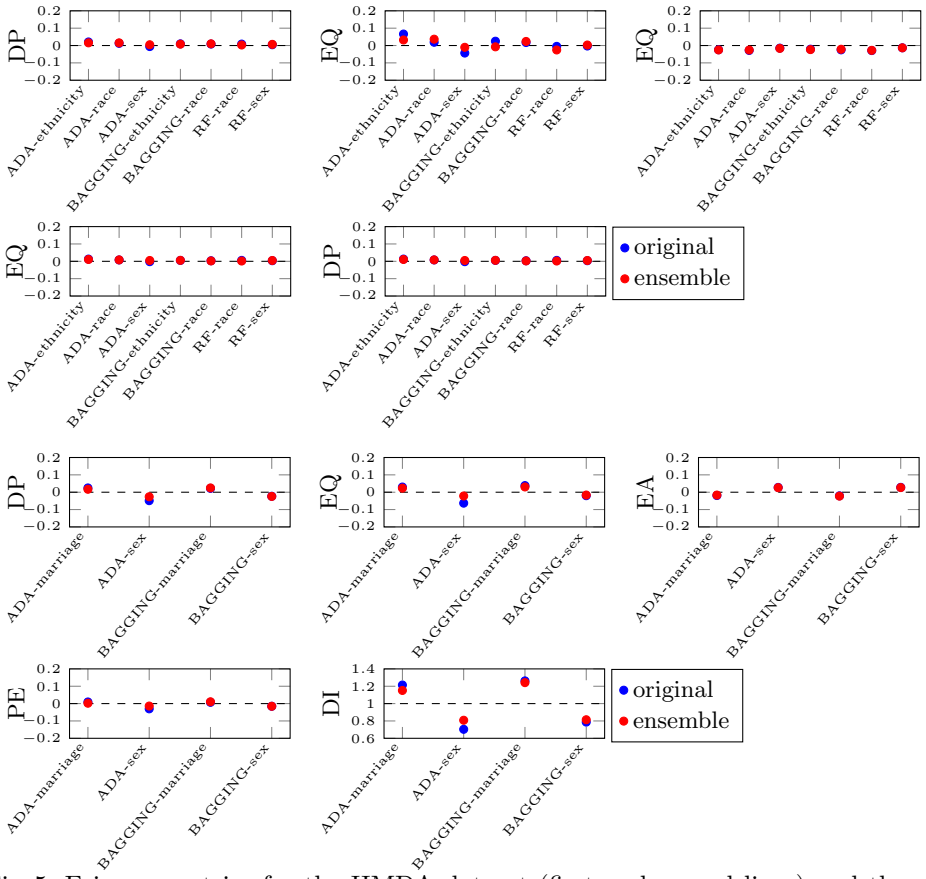

Fig. 5: Fairness metrics for the HMDA dataset (first and second lines) and the Default dataset (third and fourth lines). For both datasets, lesser original models were deemed unfair, namely, ADA, Bagging and RF on HMDA, and ADA and Bagging on Default. Even though these models were deemed unfair by *LimeOut*, most of the fairness metrics actually indicate a rather fair behaviour by the original and *LimeOut*'s ensemble models.